

## Práctica 2.6 Problemas puertas y funciones lógicas

domingo, 23 de octubre de 2022 20:11

1. En un determinado proceso industrial se verifica la calidad de unas puertas lógicas. Los chips pasan a través de tres sensores que determinan el estado de los mismos. Si al menos dos sensores detectan defectos en las mismas serán desechadas.
  - a. Escribir la **tabla de verdad** de la función de salida del detector de piezas defectuosas.
  - b. **Simplificar** la función lógica usando el método de Karnaugh.
  - c. Representa el **diagrama lógico** final del circuito desde *logic.ly*
  - d. ¿Cómo se podría implementar utilizando solo puertas lógicas **NAND**? Halla la fórmula.

### A ) TABLA DE VERDAD:

- Hay tres sensores con su estado (Con defecto: 1 – Sin defecto: 0):
  - A, B, Y C.
- Y la F es 1 si al menos dos de los sensores es 1

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

### B ) SIMPLIFICACION DE LA FUNCIÓN

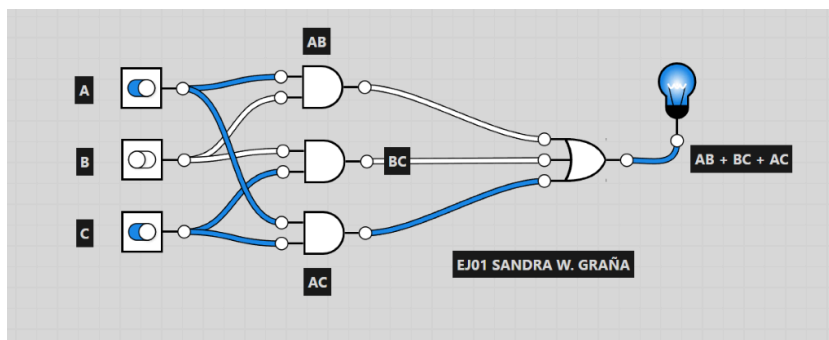
AB

		00	01	11	10
C	0	0	0	1	0
	1	0	1	1	1

A B C	A B C	A B C
1 1 0	0 1 1	1 0 1
1 1 1	1 1 1	1 1 1
AB	BC	AC

$$F = AB + BC + AC$$

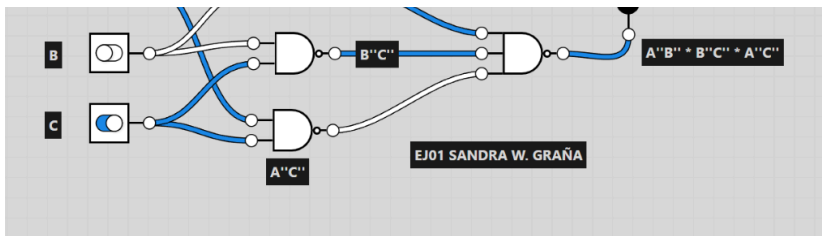
### C) REPRESENTACION LOGICA EN LOGIC.LY



### D) IMPLEMENTACION NAND

$$F = AB + BC + AC \rightarrow \text{DOBLE NEGACION EN TODO} \rightarrow F = \overline{\overline{AB} \cdot \overline{BC} \cdot \overline{AC}}$$





2. Queremos que una máquina expendedora de bebidas suministre una botella de agua cuando está pulsada la opción A, una de limonada cuando está pulsada la opción B y una bebida isotónica cuando están pulsadas ambas opciones. Por otra parte, también dispone de dos sensores C y D. El primero nos indica activándose si se ha echado la moneda correspondiente, y el segundo se activa cuando no hay botellas o latas disponibles. Si se cumplen las condiciones de suministro, un motor deberá abrir una trampilla que da acceso a la bebida.

Se pide diseñar un circuito lógico que controle el motor de apertura solucionando las siguientes cuestiones:

- Obtener la **tabla de verdad** y su función lógica.
- Simplificar** la ecuación utilizando el método apropiado.
- Representar el diagrama lógico del circuito.
- ¿Cómo se podría implementar utilizando solo puertas lógicas **NAND**? Halla la fórmula.

#### A ) TABLA DE VERDAD:

- Hay Cuatro sensores:
  - A, B, C, Y D
- Y la F es 1 si :
  - D no esta en 1
  - C esta en 1
  - A / B esta en 1 junto a C en 1

A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

#### B ) SIMPLIFICACION DE LA FUNCIÓN

AB

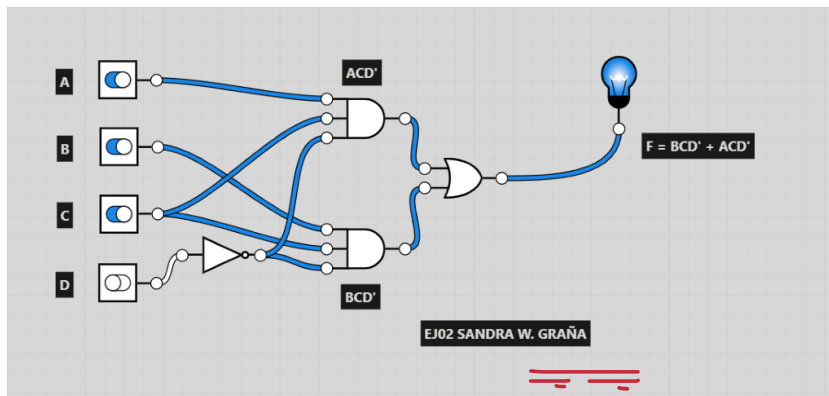
		00	01	11	10
	00	0	0	0	0
CD	01	0	0	0	0
	11	0	0	0	0
	10	0	1	1	1

A	B	C	D	A	B	C	D
0	1	1	0	1	1	1	0
1	1	1	0	1	0	1	0

BCD'	ACD'
------	------

$$F = BCD' + ACD'$$

### C) REPRESENTACION LOGICA EN LOGIC.LY



### D) IMPLEMENTACION NAND

$$F = BCD' + ACD' \rightarrow \text{DOBLE NEGACION EN TODO} \rightarrow F = BCD * ACD$$

3. Diseñe un circuito digital de control, que compare a la entrada dos palabras binarias de 2 bits (ab y cd), de manera que cuando la combinación binaria formada por los bits ab, sea mayor que la combinación binaria formada por los bits cd, la salida sea 1.
  - a. Calcule la función lógica de salida.
  - b. **Simplificar** la función lógica mediante el método de Karnaugh.
  - c. Implementar el circuito con puertas lógicas.
  - d. ¿Cómo se podría implementar utilizando solo puertas lógicas **NAND**? Halla la fórmula.

#### A ) TABLA DE VERDAD:

- Hay Cuatro sensores:
  - A, B, C, Y D
- Y la F es 1 si :
  - AB EN BINARIO > CD EN BINARIO

A	B	> < =	C	D	F
0	0	=	0	0	0
0	0	<	0	1	0
0	0	<	1	0	0
0	0	<	1	1	0
0	1	>	0	0	1
0	1	=	0	1	0
0	1	<	1	0	0
0	1	<	1	1	0
1	0	>	0	0	1
1	0	>	0	1	1
1	0	=	1	0	0
1	0	<	1	1	0
1	1	>	0	0	1
1	1	>	0	1	1
1	1	>	1	0	1
1	1	=	1	1	0

#### B ) SIMPLIFICACION DE LA FUNCIÓN

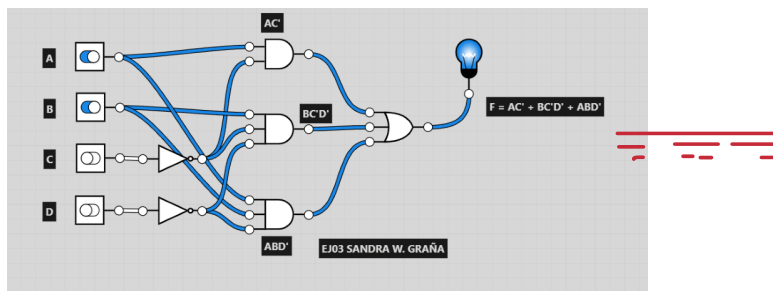
		00	01	11	10
	00	0	1	1	1

<b>CD</b>	<b>01</b>	0	0	1	1
	<b>11</b>	0	0	0	0
	<b>10</b>	0	0	1	0

<b>A B C D</b>	<b>A B C D</b>	<b>A B C D</b>
1 1 0 0	0 1 0 0	1 1 0 0
1 1 0 1	1 1 0 0	1 1 1 0
1 0 0 0		
1 0 0 1		
<b>AC'</b>	<b>BC'D'</b>	<b>ABD'</b>

$$F = AC' + BC'D' + ABD'$$

### C) REPRESENTACION LOGICA EN LOGIC.LY



### D) IMPLEMENTACION NAND

$$F = F = AC' + BC'D' + ABD' \rightarrow \text{DOBLE NEGACION EN TODO} \rightarrow F = AC + BCD + ABD$$